UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/989,777	11/19/2001	Craig Nemecek	CYPR-CD01208M	2046

7590 03/13/2007

WAGNER, MURABITO & HAO LLP
Third Floor
Two North Market Street
San Jose, CA 95113

EXAMINER	
SHARON, AYAL I	
ART UNIT	**PAPER NUMBER**
2123	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	03/13/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)	
	09/989,777	NEMECEK, CRAIG	
	Examiner	Art Unit	
	Ayal I. Sharon	2123	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *12/18/2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-28* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-28* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *2/10/2006* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Introduction

1. Claims 1-28 of U.S. Application 09/989,777 are currently pending.
2. The application was originally filed on filed on 11/19/2001.
3. This action is non-final.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

5. The prior art used for these rejections is as follows:
    a. U.S. Patent 7,089,175 to Nemecek et al. (Henceforth referred to as "**Nemecek et al.**").
6. The claim rejections are hereby summarized for Applicant's convenience. The detailed rejections follow.
7. **Claims 1-28 are rejected under 35 U.S.C. 102(e) as being anticipated by Nemecek et al.**

8. In regards to Claim 1, Nemecek et al. teaches the following limitations:

> 1. A method for performing a sleep operation in a system that includes a device under test and an emulator device, said method comprising:
>
> a) executing instructions on said device under test;

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

> b) emulating the functions of said device under test by operating said emulator device in lock-step fashion with said device under test; and

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

> c) performing a sleep operation, comprising:

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

> c1) upon receiving a first signal that indicates that a sleep function is to be performed, initiating said sleep function at said device under test;

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

> c2) turning off one or more clock of said device under test; and

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

> c3) discontinuing execution of instructions that are performed in lock-step by said emulator device upon turning off said clock.

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

9. In regards to Claim 2, Nemecek et al. teaches the following limitations:

> 2. The method of Claim 1 wherein said clock comprises an internal CPU clock.

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

10. In regards to Claim 3, Nemecek et al. teaches the following limitations:

3. The method of Claim 2 wherein said first signal is generated by said device under test and is transmitted internally to a register that indicates that a sleep function is to be performed.

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

11. In regards to Claim 4, Nemecek et al. teaches the following limitations:

4. The method of Claim 1 further comprising:

when said sleep function has been completed by said device under test, turning on said clock and sending a second signal from said device under test to said emulator device;

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

receiving said second signal at said emulator device;

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

determining the number of clock signals received at said emulator device since said second signal was received; and

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

resuming execution of said instructions that are performed in lock-step at said emulator device when said determined number of clock signals received at said emulator device since said second signal was received equals a predetermined value.

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

12. In regards to Claim 5, Nemecek et al. teaches the following limitations:

5. The method of Claim 4 wherein said device under test further comprises a microcontroller and wherein said first signal comprises a first bit, said first bit received at a register of said microcontroller to indicate that a sleep function is to be performed.

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

13. In regards to Claim 6, Nemecek et al. teaches the following limitations:

6. The method of Claim 5 wherein said emulator device further comprises a Field Programmable Gate Array (FPGA) device.

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

14. In regards to Claim 7, Nemecek et al. teaches the following limitations:

7. A method for performing a stall operation in a system that includes a device under test and an emulator device, said method comprising:

a) executing instructions on said device under test;

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

b) emulating the functions of said device under test by operating said emulator device in lock-step fashion with said device under test; and

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

c) performing a stall operation, comprising:

(See Nemecek et al., especially: col.10, lines 29-43 which refers to "halt" and "break" signals; and also col.13, lines 6-53 which refers to "break", "stop", "interrupt", and "reset")

c1) said device under test conveying clock signals to said emulator device;

(See Nemecek et al., especially: col.10, lines 29-43 which refers to "halt" and "break" signals; and also col.13, lines 6-53 which refers to "break", "stop", "interrupt", and "reset")

c2) upon receiving a first signal that indicates that a stall function is to be performed, initiating said stall function at said device under test;

(See Nemecek et al., especially: col.10, lines 29-43 which refers to "halt" and "break" signals; and also col.13, lines 6-53 which refers to "break", "stop", "interrupt", and "reset")

c3) upon receiving said first signal, discontinuing said sending of said clock signals from said device under test to said emulator device; and

(See Nemecek et al., especially: col.10, lines 29-43 which refers to "halt" and "break" signals; and also col.13, lines 6-53 which refers to "break", "stop", "interrupt", and "reset")

c4) discontinuing execution of said instructions that are performed in lock-step at said emulator device while said sending of said clock signals is discontinued.

(See Nemecek et al., especially: col.10, lines 29-43 which refers to "halt" and "break" signals; and also col.13, lines 6-53 which refers to "break", "stop", "interrupt", and "reset")

15. In regards to Claim 8, Nemecek et al. teaches the following limitations:

> 8. The method according to claim 7 wherein said device under test is a microcontroller and wherein said emulator device includes a field programmable gate array (FPGA), said clock signals further comprising signals from said microcontroller central processing unit clock.

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

16. In regards to Claim 9, Nemecek et al. teaches the following limitations:

> 9. The method of Claim 8 further comprising: resuming sending of said clock signals from said device under test to said emulator device when said stall function has been completed by said device under test, said emulator device operable upon receiving said clock signals to resume execution of said instructions that are performed in lock-step.

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

17. In regards to Claim 10, Nemecek et al. teaches the following limitations:

> 10. A method for performing a sleep operation, comprising:
>
> executing a sequence of instructions by a device under test, said device under test including at least one clock for generating clock signals;

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

> executing said sequence of instructions by an emulator device emulating the functions of said device under test, said emulator device executing said sequence of instructions in lock-step fashion with said device under test;

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

> receiving a first signal at a register of said device under test that indicates that a sleep function is to be initiated;

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

> initiating said sleep function at said device under test upon receipt of said first signal;

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

turning off said at least one clock of said device under test; and

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

discontinuing execution of instructions that are performed in lock-step by

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

said emulator device upon said turning off of said clock.

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

18. In regards to Claim 11, Nemecek et al. teaches the following limitations:

11. The method according to claim 10 wherein said device under test is a microcontroller and wherein said emulator device includes a field programmable gate array (FPGA).

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

19. In regards to Claim 12, Nemecek et al. teaches the following limitations:

12. The method of Claim 11 wherein said at least one clock includes a microcontroller CPU clock.

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

20. In regards to Claim 13, Nemecek et al. teaches the following limitations:

13. The method of Claim 12 further comprising:

when said sleep function has been completed by said device under test, resuming generation of clock signals at said device under test and coupling said clock signals to said emulator device;

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

when said sleep function has been completed by said device under test, sending a second signal from said device under test to said emulator device;

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

receiving said second signal at said emulator device;

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

determining the number of clock signals received at said emulator device since said second signal was received; and

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

resuming execution of said instructions that are performed in lock-step at said emulator device when said determined number of clock signals received at said emulator device since said second signal was received equals a predetermined value.

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

21. In regards to Claim 14, Nemecek et al. teaches the following limitations:

14. The method according to claim 13 wherein said device under test is a microcontroller and wherein said emulator device includes a field programmable gate array (FPGA).

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

22. In regards to Claim 15, Nemecek et al. teaches the following limitations:

15. The method of Claim 14 wherein said first signal is a first bit, said sleep function initiated upon the receipt of said first bit at a register of said microcontroller.

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

23. In regards to Claim 16, Nemecek et al. teaches the following limitations:

16. A method for performing a stall operation, comprising:

executing a sequence of instructions by a device under test;

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

executing said sequence of instructions by an emulator device emulating the functions of said device under test, said emulator device executing said sequence of instructions in lock-step fashion with said device under test;

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

said device under test sending clock signals to said emulator device; receiving a first signal at a register of said device under test that indicates that a stall function is to be initiated;

(See Nemecek et al., especially: col.10, lines 29-43 which refers to "halt" and "break" signals; and also col.13, lines 6-53 which refers to "break", "stop", "interrupt", and "reset")

initiating said stall function at said device under test upon receipt of said first signal;

(See Nemecek et al., especially: col.10, lines 29-43 which refers to "halt" and "break" signals; and also col.13, lines 6-53 which refers to "break", "stop", "interrupt", and "reset")

discontinuing said sending of said clock signals from said device under test to said emulator device upon initiation of a stall function at said device under test; and

(See Nemecek et al., especially: col.10, lines 29-43 which refers to "halt" and "break" signals; and also col.13, lines 6-53 which refers to "break", "stop", "interrupt", and "reset")

discontinuing execution of said sequence of instructions at said emulator device while said sending of said clock signals is discontinued.

(See Nemecek et al., especially: col.10, lines 29-43 which refers to "halt" and "break" signals; and also col.13, lines 6-53 which refers to "break", "stop", "interrupt", and "reset")

24. In regards to Claim 17, Nemecek et al. teaches the following limitations:

17. The method according to claim 16 wherein said device under test is a microcontroller and wherein said emulator device includes a field programmable gate array (FPGA).

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

25. In regards to Claim 18, Nemecek et al. teaches the following limitations:

18. The method according to Claim 17 wherein said clock signals further comprise signals from a central processing unit clock of said microcontroller.

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

26. In regards to Claim 19, Nemecek et al. teaches the following limitations:

> 19. The method of Claim 18 further comprising: resuming sending of said clock signals from said device under test to said emulator device when said stall function has been completed by said device under test, said emulator device operable upon receiving said clock signals to resume execution of said sequence of instructions.

(See Nemecek et al., especially: col.10, lines 29-43 which refers to "halt" and "break" signals; and also col.13, lines 6-53 which refers to "break", "stop", "interrupt", and "reset")

27. In regards to Claim 20, Nemecek et al. teaches the following limitations:

> 20. The method of Claim 19 wherein said sequence of instructions comprises the core processing functions of said microcontroller.

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

28. In regards to Claim 21, Nemecek et al. teaches the following limitations:

> 21. An in-circuit emulation system comprising:
>
> a device under test that executes a sequence of instructions, said device under test operable, upon receiving a first signal, to initiate a stall function;

(See Nemecek et al., especially: col.10, lines 29-43 which refers to "halt" and "break" signals; and also col.13, lines 6-53 which refers to "break", "stop", "interrupt", and "reset")

> an emulator device for emulating the functions of said device under test, said emulator device operable so as to execute said sequence of instructions in lock-step fashion with said device under test, said emulator device configured for receiving clock signals sent by said device under test; and

(See Nemecek et al., especially: col.10, lines 29-43 which refers to "halt" and "break" signals; and also col.13, lines 6-53 which refers to "break", "stop", "interrupt", and "reset")

> wherein said device under test sends clock signals to said emulator device, said device under test operable, upon receiving said first signal, to discontinue sending said clock signals to said emulator device, and said emulator device operable, upon said discontinuation of said clock signals from said device under test, to discontinue execution of said sequence of instructions.

(See Nemecek et al., especially: col.10, lines 29-43 which refers to "halt" and "break" signals; and also col.13, lines 6-53 which refers to "break", "stop", "interrupt", and "reset")

29. In regards to Claim 22, Nemecek et al. teaches the following limitations:

> 22. The in-circuit emulation system of Claim 21 wherein said device under test is a microcontroller, said microcontroller operable to resume sending said clock signals to said emulator device when said stall function has been completed by said microcontroller, said emulator device operable upon receiving said clock signals to resume execution of said sequence of instructions.

(See Nemecek et al., especially: col.10, lines 29-43 which refers to "halt" and "break" signals; and also col.13, lines 6-53 which refers to "break", "stop", "interrupt", and "reset")

30. In regards to Claim 23, Nemecek et al. teaches the following limitations:

> 23. The in-circuit emulation system of Claim 22 wherein said clock signals further comprise signals from a central processing unit clock of said microcontroller.

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

31. In regards to Claim 24, Nemecek et al. teaches the following limitations:

> 24. The in-circuit emulation system of Claim 23 wherein said emulator device comprises a field programmable gate array (FPGA).

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

32. In regards to Claim 25, Nemecek et al. teaches the following limitations:

> 25. An in-circuit emulation system comprising;

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

> a device under test that executes a sequence of instructions, said device under test operable, upon receiving a first signal, to initiate a sleep function at said device under test and to turn off a clock of said device under test; and

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

> an emulator device for emulating the functions of said device under test, said emulator device operable so as to execute said sequence of instructions in lock-step fashion with said device under test, said emulator device operable, upon said turning off

of said clock to discontinue execution of said sequence of instructions at said emulator device.

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

33. In regards to Claim 26, Nemecek et al. teaches the following limitations:

> 26. The in-circuit emulation system of Claim 25 wherein said device under test comprises a microcontroller, said device under test operable when said sleep function has been completed by said device under test to turn on said at least one clock and to send a second signal to said emulator device, said emulator device operable upon receiving said second signal to determine the number of clock signals received at said emulator device since said second signal was received and said emulator device operable to resume execution of said sequence of instructions when said determined number of clock signals received at said emulator device since said second signal was received equals a predetermined value.

(See Nemecek et al., especially: col.3, lines 25-32 and 43-48; Figs. 9 and 10 and associated text at col.16, line 11 to col.17, line 5)

34. In regards to Claim 27, Nemecek et al. teaches the following limitations:

> 27. The in-circuit emulation system of Claim 26 wherein 22 device under test is a microcontroller, said at least one clock further comprising a central processing unit clock of said microcontroller.

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

35. In regards to Claim 28, Nemecek et al. teaches the following limitations:

> 28. The in-circuit emulation system of Claim 27 wherein said emulator device comprises a field programmable gate array (FPGA).

(See Nemecek et al., especially: col.8, line 48 to col.9, line 6; col.10, line 51 to col.11, line 41)

Response to Arguments

Re: Claim Rejections - 35 USC § 102

36. Examiner finds applicant's remarks (filed 12/18/2006) to be unpersuasive:

37. In response to Applicant's arguments presented in Applicant's response filed 12/18/2006, the Examiner refers to the following selected sentences from col.16 of Nemecek (emphasis added):

> In order to carry out programming functions on a microcontroller mounted in socket 620, the operation of microcontroller 232 should be placed in a mode such that it does not disturb the programming process, i.e., a sleep mode. This is accomplished by placing microcontroller 232 in a sleep mode while simultaneously placing the microcontroller 620 in a programming mode. Programming date can then be clocked into the microcontroller in socket 620 from base station 218 without the operation being disrupted by the microcontroller 232.
>
> In order to accomplish both debug operations and programming operations using pod assembly 610, microcontroller 232 can be selectively placed in either a debug or a sleep mode.
>
> FIG. 10 depicts a process 800 used to place the microcontroller 232 in a sleep mode so that it does not disrupt programming operations of the microcontroller in socket 620. Of course, the microcontroller in socket 620 is placed in a programming mode while the microcontroller 232 is in a sleep mode in order to effect programming of the microcontroller in socket 620. Of course, when microcontroller 232 is in the debug mode, there is normally no microcontroller present in socket 620 to disrupt the debug operation.
>
> Process 800 starts at 804 after which power is turned on to the pod assembly at 808. The In-Circuit Emulation system holds power and reset asserted at 812 and watches for the data0 line to go to a logic high indicating stable power at 816. At 820 the In-Circuit Emulation system pulls the data0 line to a logic low at 820 and then releases the reset line while holding the data0 line at a logic low at 824. **This sequence of events causes the microcontroller 232 to enter the sleep mode.**
>
> The In-Circuit Emulation system then uses the data1 line as a programming clock line in order to clock data into the microcontroller in socket 620 at 824. **Prior to clocking in programming data, a key code is clocked in at 828.** This key code is specific to the particular microcontroller device being programmed in order to prevent unauthorized modifications of the program. **If the key code does not match at 832, the microcontroller in socket 620 begins running whatever code is stored internally at 840. If the key matches at 832, the**

> **microcontroller in socket 620 enters a program mode so that it can accept program code.** Data representing program code are then clocked into the programmable microcontroller in socket 620 at 848 by applying clock signals to the data1 line and data to the data0 line in order to program the socketed microcontroller being programmed. **This process proceeds until all of the program lines have been clocked in at 852** after which the microcontroller halts and power is turned off to the pod at 856. The programming process is now completed at 860 and the microcontroller in socket 620 can be removed for use.

38. On p.13 of Applicant's remarks, Applicant argues that Nemecek does not "teach or suggest turning off one or more clock of the device under test" as claimed in Claim 1. Yet Nemecek expressly teaches in lines 1-10 of col.16 that once the microcontroller 232 is in sleep mode, "it does not disturb the programming process." Moreover, Nemecek teaches in lines 30-34 of col.16 that "FIG. 10 depicts a process 800 used to place the microcontroller 232 in a sleep mode so that it does not disrupt programming operations of the microcontroller in socket 620."

According to the Microsoft Press Computer User's Dictionary, "sleep" has two computer-related definitions: the verb means to "suspend[ing] operation without terminating", while the noun means "a temporary state of suspension during which a process remains in memory, so that some event, such as an interrupt or a call from another process, can 'awaken' it."

Suspending processes can be done in software (in an operating system) or in hardware (by turning off a clock). Nemecek is completely silent about the use of an operating system to activate the sleep mode in the device under test, and instead indicates that the triggers for the sleep and debug modes are

hardware signals (see col.16, lines 42-47 and col.17, lines 6-15). Examiner therefore finds it inherent that Nemecek places the device in sleep mode by hardware means (turning off a clock).

39. On p.17 of Applicant's remarks, Applicant argues that Nemecek does not "teach or suggest discontinuing the sending of the clock signals" as claimed in Claim 7. Yet Nemecek expressly teaches in lines 1-10 of col.16 that once the microcontroller 232 is in sleep mode, "it does not disturb the programming process." Moreover, Nemecek teaches in lines 30-34 of col.16 that "FIG. 10 depicts a process 800 used to place the microcontroller 232 in a sleep mode so that it does not disrupt programming operations of the microcontroller in socket 620."

Moreover, according to the Microsoft Press Computer User's Dictionary, "sleep" has two computer-related definitions: the verb means to "suspend[ing] operation without terminating", while the noun means "a temporary state of suspension during which a process remains in memory, so that some event, such as an interrupt or a call from another process, can 'awaken' it."

Examiner therefore respectfully disagrees with Applicant, and interprets Nemecek as teaching that when once the microcontroller 232 is in sleep mode, it does not send any signals to the microcontroller in socket 620.

40. On p.14 of Applicant's remarks, Applicant argues that Nemecek does not "teach or suggest determining the number of clock signals and resuming execution in

lock step" as claimed in Claims 4, 13, and 26. Yet Nemecek expressly teaches the following in col.16, lines 50-58:

> Prior to clocking in programming data, a key code is clocked in at 828. This key code is specific to the particular microcontroller device being programmed in order to prevent unauthorized modifications of the program. If the key code does not match at 832, the microcontroller in socket 620 begins running whatever code is stored internally at 840. If the key matches at 832, the microcontroller in socket 620 enters a program mode so that it can accept program code.

Examiner interprets that this "key code" corresponds to the claimed "number of clock signals received ... [that] equals a predetermined value.

Conclusion

41. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

42. A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Correspondence Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ayal I. Sharon whose telephone number is (571) 272-3714. The examiner can normally be reached on Monday through Thursday, and the first Friday of a biweek, 8:30 am – 5:30 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Paul Rodriguez can be reached at (571) 272-3753.

Any response to this office action should be faxed to (703) 872-9306, or mailed to:

USPTO
P.O. Box 1450
Alexandria, VA 22313-1450

or hand carried to:

USPTO
Customer Service Window
Randolph Building
401 Dulany Street
Alexandria, VA 22314

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the Tech Center 2100 Receptionist, whose telephone number is (571) 272-2100.

Ayal I. Sharon
Art Unit 2123
March 8, 2007

3/9/07

PAUL RODRIGUEZ
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100